

# **INFORME INICIAL TFG: DIGITALIZACIÓN DE UN JUEGO**

**Tutor: Miguel Hernández Cabronero**  
**Curso 2024-25**

Leyva García, Pau - 1636526

<b>1. Introducción.....</b>	<b>3</b>
<b>2. Estado del arte.....</b>	<b>3</b>
<b>3. Objetivos del proyecto.....</b>	<b>4</b>
<b>4. Casos de uso.....</b>	<b>5</b>
CU01 - Crear una partida.....	5
CU02 - Introducir palabras al juego.....	5
CU03 - Desarrollo del juego.....	5
CU04 - Cambio de ronda.....	6
CU05 - Finalizar la partida y mostrar resultados.....	6
CU06 - Pausar y editar el cronómetro.....	6
<b>5. Diagrama de arquitectura y despliegue del sistema.....</b>	<b>7</b>
<b>6. Diagrama de Gantt.....</b>	<b>7</b>
<b>7. Bibliografía.....</b>	<b>9</b>

# 1. Introducción

Este Trabajo de Fin de Grado consiste en digitalizar un juego tradicional en el que los participantes escriben palabras aleatorias, las mezclan, se forman equipos y, en turnos que funcionan de manera rotativa entre todos los jugadores, un representante de cada equipo debe conseguir que sus compañeros adivinen la mayor cantidad de palabras posibles en cierto tiempo, ya sea mediante una descripción completa, mímica u otra temática dependiendo de la ronda en curso. Así el equipo que consiga adivinar más palabras al finalizar todas las rondas será el ganador.

Se trata de un juego bastante conocido entre los grupos de amigos, pero su versión clásica presenta ciertas limitaciones logísticas, como la necesidad de contar con papel, bolígrafos y un recipiente adecuado para mezclar las palabras. Dificultades que pueden llevar a que la preparación sea lenta o incluso impedir que se pueda jugar.

Con el desarrollo de esta aplicación web, se busca eliminar estas barreras y permitir que los jugadores puedan disfrutar del juego de manera más ágil y accesible desde cualquier dispositivo sin necesidad de materiales físicos.

## 2. Estado del arte

Actualmente existen varias aplicaciones que digitalizan juegos tradicionales, como por ejemplo el “Monopoly” o el “Heads Up” y otros de un carácter más similar como el “Tabú”, pero no se encuentra ninguna aplicación que lo haga para este juego en concreto.

Para empezar con el desarrollo, la primera elección crucial ha sido decidir desarrollar una aplicación web en lugar de una app móvil, debido a la necesidad de accesibilidad y facilidad de uso. Una aplicación web permite a los usuarios jugar desde cualquier dispositivo con un navegador compatible, sin necesidad de descargar una aplicación, facilitando así el acceso inmediato al juego.

Para el frontend, se ha seleccionado Vue.js, tecnología que permite realizar un desarrollo rápido y eficiente para las interfaces de usuario interactivas y responsivas, para así conseguir una buena experiencia de usuario. Además, aporta mantenibilidad y escalabilidad al proyecto por la creación de componentes reutilizables. También se han considerado otras tecnologías muy relevantes como React.js y Angular, pero para este proyecto se ha preferido contar con Vue.js por su simplicidad, estructura y beneficios que aporta.

En cuanto al backend, la opción escogida ha sido la combinación de Node.js con Express.js, ya que cuenta con un amplio ecosistema de paquetes y herramientas que pueden integrarse fácilmente, ayudando a la creación de APIs y servidores web. Otras tecnologías consideradas fueron usar el framework Django de Python o Spring Boot, pero Node.js aporta una gran flexibilidad y rendimiento en este caso.

Para la base de datos, se ha considerado tanto PostgreSQL como SQLite. Aunque PostgreSQL es una opción robusta y capaz de manejar grandes volúmenes de datos y múltiples usuarios concurrentes de manera eficiente, se ha decidido valorar y probar SQLite, ya que proporciona un rendimiento adecuado para aplicaciones con bajo volumen de datos y pocas operaciones concurrentes.

Respecto al despliegue, se ha optado por utilizar una máquina virtual o un PC local. De esta forma se obtiene un control total sobre el entorno de ejecución, permitiendo configuraciones personalizadas y ajustes específicos según las necesidades del proyecto. Al no depender de servicios externos como Vercel, Netlify, Railway o Render, se evita la dependencia de terceros y se garantiza que la aplicación puede ser ejecutada en cualquier entorno controlado.

### 3. Objetivos del proyecto

El principal objetivo de este Trabajo de Fin de Grado es desarrollar una aplicación web interactiva que consiga digitalizar un juego tradicional basado en adivinar palabras por equipos, permitiendo a todos los usuarios una experiencia intuitiva y accesible. Para ello, se han definido una serie de objetivos:

1. Implementar las mismas mecánicas principales del juego físico para que no cambie la forma de jugar y la transición hacia el entorno digital sea lo más cómoda posible. Por tanto, deberían estar las funcionalidades de crear equipos, introducir palabras, sistema de turnos, cronómetro y puntuaciones lo más acordes al juego tradicional.
2. Permitir a los jugadores participar únicamente con un dispositivo con acceso a un navegador web. De esta forma podemos deshacernos de las necesidades de materiales físicos, ya sea papel, bolígrafos, recipientes para mezclar las palabras, etc.
3. Diseñar una UI/UX intuitiva, de manera que tanto usuarios con experiencia en el juego como usuarios que nunca han jugado puedan entender el funcionamiento y jugar sin necesidad de instrucciones extensas.
4. Garantizar que la aplicación sea compatible con distintos dispositivos, ya que los usuarios pueden usar móviles pero también, por ejemplo, tablets.
5. Desplegar la aplicación desde una Máquina Virtual o PC propio para no depender de nada externo.
6. Realizar un backend que permita gestionar las mecánicas del juego.
7. Permitir la configuración personalizada de partidas, incluyendo tanto la creación del número de equipos necesarios, los nombres de los participantes de cada equipo y el orden de turnos.
8. Permitir a cada usuario introducir las palabras que quiera añadir al juego, para después darles un orden aleatorio para jugar con ellas.
9. Incluir un sistema de cronómetro que facilite el control del tiempo en cada ronda.
10. Implementar un sistema de puntuación para cada equipo según el número de palabras acertadas (sumando un punto por cada una) o pasadas (descontando un punto por cada una) para llevar un registro de quién va ganando.

11. Probar la aplicación en distintos dispositivos y navegadores para detectar posibles problemas de compatibilidad.
12. Recoger feedback de usuarios a través de pruebas piloto para mejorar la usabilidad y corregir posibles errores antes de la entrega final.

## 4. Casos de uso

### CU01 - Crear una partida

Un usuario crea una nueva partida, estableciendo el número de equipos y los nombres de los jugadores.

1. El usuario introduce los equipos y los participantes.
2. Se elige el número de rondas y se puede introducir el título/temática de cada una.
3. Se genera el orden de turnos rotativo dentro de los equipos y entre los equipos.

### CU02 - Introducir palabras al juego

Cada usuario introduce una serie de palabras que se utilizarán durante la partida.

1. Se indica en pantalla qué jugador debe introducir palabras.
2. Cada jugador introduce un número determinado de palabras.
3. Cuando todos han introducido palabras, se puede comenzar la partida.

### CU03 - Desarrollo del juego

Inicia una ronda, mostrando el jugador que le toca jugar y se desarrolla la ronda hasta acertar todas las palabras.

1. Se muestra en pantalla la ronda actual, el título y el jugador que tiene el turno.
2. Al iniciar el turno, comienza el cronómetro.
3. Aparece una palabra con opciones para marcarla como acertada (+1 punto) o pasarla (-1 punto).
4. Una vez se acaba el tiempo, el turno pasa al siguiente jugador.
5. El juego continúa hasta que se hayan acertado todas las palabras.

## CU04 - Cambio de ronda

Al finalizar una ronda, se muestra la puntuación hasta el momento de cada equipo y se da la opción de siguiente ronda si aún falta jugar alguna.

1. Cuando todas las palabras han sido acertadas, se muestra la puntuación.
2. Se da la opción de pasar a la siguiente ronda.
3. Se repite el proceso hasta completar todas las rondas.

## CU05 - Finalizar la partida y mostrar resultados

Una vez finalizadas todas las rondas, el sistema muestra la puntuación final de los equipos.

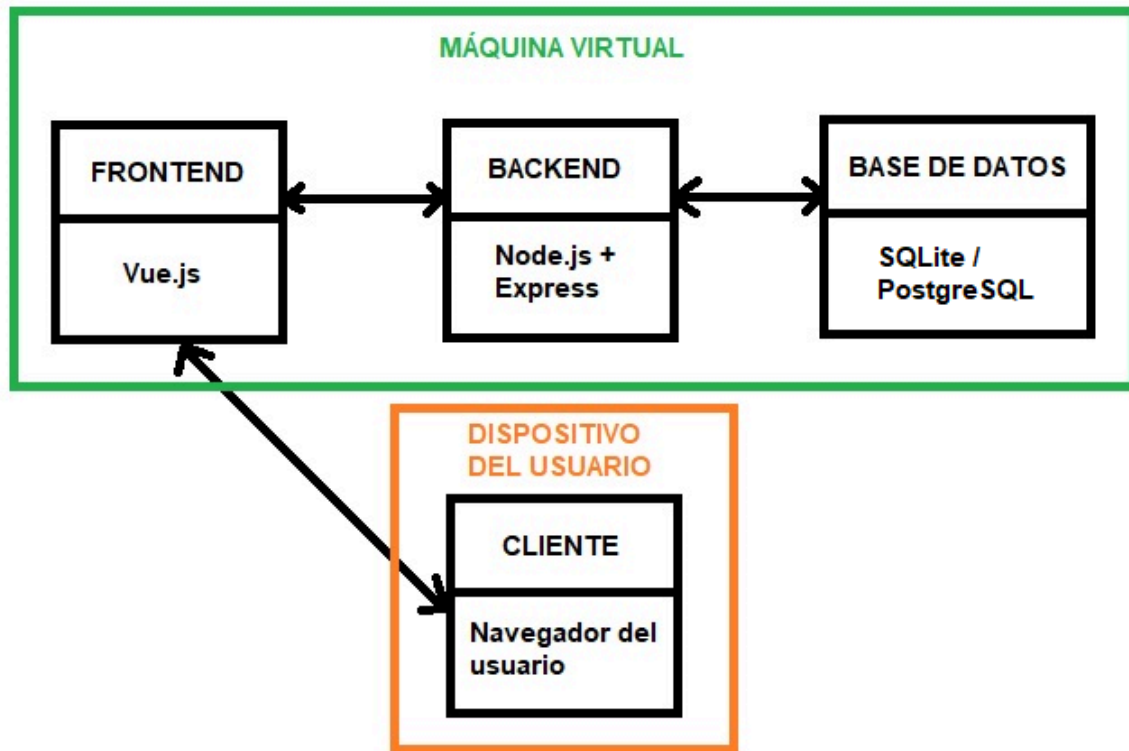
1. Se muestran las puntuaciones finales y el equipo ganador.
2. Se ofrece la opción de volver a jugar.

## CU06 - Pausar y editar el cronómetro

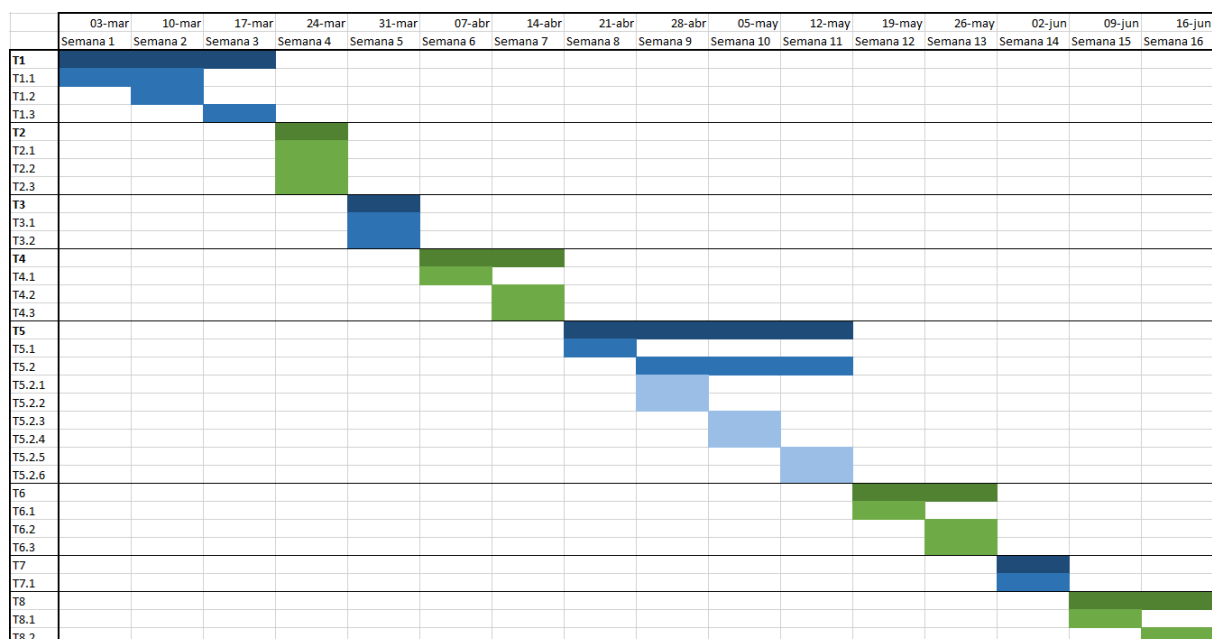
Durante una ronda, el jugador puede pausar el cronómetro o modificar el tiempo restante.

1. El jugador accede al control del cronómetro.
2. Puede pausar, reiniciar o editar el tiempo restante.
3. Guarda los cambios y continúa con la ronda.

## 5. Diagrama de arquitectura y despliegue del sistema



## 6. Diagrama de Gantt



- T1: Diseño de la UI/UX → Marzo (3-23) → Semanas 1,2,3
- T1.1: Diseño de la interfaz de usuario. → Semana 1,2
  - T1.2: Creación de wireframes y prototipos. → Semana 2,3
  - T1.3: Pruebas de usabilidad y ajustes. → Semana 2,3
- T2: Configuración del entorno de desarrollo → Marzo (24-30) → Semana 4
- T2.1: Configuración de la máquina virtual o PC local. → Semana 4
  - T2.2: Instalación de herramientas y dependencias. → Semana 4
  - T2.3: Configuración del repositorio de código. → Semana 4
- T3: Desarrollo del frontend → Abril (1-6) → Semana 5
- T3.1: Implementación de la estructura básica de la aplicación con Vue.js. → Semana 5
  - T3.2: Creación de componentes reutilizables. → Semana 5
- T4: Desarrollo del backend → Abril (7-20) → Semanas 6,7
- T4.1: Configuración de Node.js con Express.js. → Semana 6
  - T4.2: Probar SQLite y PostgreSQL. → Semana 7
  - T4.3: Integración con la base de datos. → Semana 7
- T5: Integración frontend y backend → Abril (21-30) y Mayo (1-18) → Semanas 8,9,10,11
- T5.1: Conexión del frontend con el backend. → Semana 8
  - T5.2: Implementación de la lógica de juego: → Semanas 9,10,11
    - T5.2.1: CU01 - Crear una partida. → Semana 9
    - T5.2.2: CU02 - Introducir palabras al juego. → Semana 9 (después de T5.2.1)
    - T5.2.3: CU03 - Desarrollo del juego. → Semana 10 (después de T5.2.2)
    - T5.2.4: CU04 - Cambio de ronda. → Semana 10
    - T5.2.5: CU05 - Finalizar la partida y mostrar resultados. → Semana 11 (después de T5.2.4)
    - T5.2.6: CU06 - Pausar y editar el cronómetro → Semana 11
- T6: Pruebas y depuración → Mayo (19-31) y Junio (1) → Semanas 12,13
- T6.1: Pruebas funcionales y de usabilidad. → Semana 12
  - T6.2: Identificación y corrección de errores. → Semana 13
  - T6.3: Pruebas de compatibilidad en distintos dispositivos y navegadores. → Semana 13
- T7: Despliegue → Junio (2-8) → Semana 14
- T7.1: Despliegue de la aplicación en la máquina virtual o PC local. → Semana 14
- T8: Documentación → Junio (9-22) → Semana 15,16
- T8.1: Redacción de la documentación técnica y del informe final. → Semana 15
  - T8.2: Preparación para la presentación y entrega del TFG. → Semana 16



## 7. Bibliografía

- [1] Express.js, "Express - Node.js web application framework." [En línea]. Disponible en: <https://expressjs.com/> (consultado en febrero de 2025).
- [2] Kinsta, "13 bibliotecas Node.js para potenciar tus proyectos," 2023. [En línea]. Disponible en: <https://kinsta.com/es/blog/bibliotecas-node-js/> (consultado en febrero de 2025).
- [3] Kinsta, "¿Qué es Express?" [En línea]. Disponible en: <https://kinsta.com/es/base-de-conocimiento/que-es-express/> (consultado en febrero de 2025).
- [4] Node.js, "Node.js Documentation." [En línea]. Disponible en: <https://nodejs.org/> (consultado en febrero de 2025).
- [5] PostgreSQL, "PostgreSQL Official Website." [En línea]. Disponible en: <https://www.postgresql.org/> (consultado en febrero de 2025).
- [6] SQLite, "SQLite Documentation." [En línea]. Disponible en: <https://www.sqlite.org/> (consultado en febrero de 2025).
- [7] Vue.js, "Vue.js Documentation." [En línea]. Disponible en: <https://vuejs.org/> (consultado en febrero de 2025).
- [8] Wikipedia, "SQLite." [En línea]. Disponible en: <https://es.wikipedia.org/wiki/SQLite> (consultado en febrero de 2025).
- [9] Lingolex, "Tabú - Juego de palabras." [En línea]. Disponible en: <https://lingolex.com/juegos/taboo/index.php> (consultado en febrero de 2025).
- [10] Charades.app, "Juego de mímica online." [En línea]. Disponible en: <https://charades.app/es/> (consultado en febrero de 2025).
- [11] YouTube, "Deploy to Railway, FlyIO, or Render with Ruby on Rails," 2024. [En línea]. Disponible en: [https://www.youtube.com/watch?v=PsS0bF\\_xmXQ](https://www.youtube.com/watch?v=PsS0bF_xmXQ) (consultado en febrero de 2025).
- [12] YouTube, "Vercel vs Netlify - Which One Should You Use?," 2023. [En línea]. Disponible en: <https://www.youtube.com/watch?v=2bqSIVtF4jc> (consultado en febrero de 2025).
- [13] Cypress.io, "Cypress Documentation." [En línea]. Disponible en: <https://docs.cypress.io/> (consultado en marzo de 2025).